

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-149425

(43)Date of publication of application : 24.05.2002

(51)Int.Cl.

G06F 9/46
G06F 12/00

(21)Application number : 2001-287441

(71)Applicant : MICROSOFT CORP

(22)Date of filing : 20.09.2001

(72)Inventor : CLIFT NEILL

(30)Priority

Priority number : 2000 738568

Priority date : 15.12.2000

Priority country : US

2000 244314

30.10.2000

US

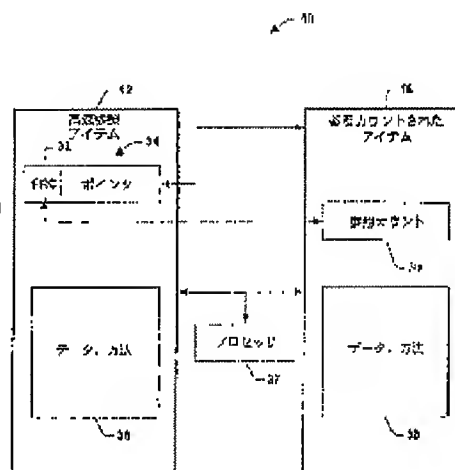
(54) SYSTEM AND METHOD FOR QUICKLY REFERRING TO REFERENCE COUNTED ITEM

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a system and method for promoting the performance of a multi-thread application and/or system using parallel processors by quickly referring to a reference counted item.

SOLUTION: In this system, a reference counted item is established, and a quick reference item is established, and a pointer to the reference counted item is set, and the local copy of the reference counted item is stored in the pointer to the reference counted item, and the pointer is related with the quick reference item.

Therefore, in this system and method, reference to the reference counted item can be acquired from the quick reference item without using any lock algorithm for locking, accessing, and unlocking the reference counted item.



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2002-149425
(P2002-149425A)

(43) 公開日 平成14年5月24日 (2002.5.24)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード* (参考)
G 0 6 F 9/46	3 6 0	G 0 6 F 9/46	3 6 0 C 5 B 0 6 0
	3 4 0		3 4 0 F 5 B 0 9 8
12/00	5 7 2	12/00	5 7 2 A
	5 9 1		5 9 1

審査請求 未請求 請求項の数37 O L (全 25 頁)

(21) 出願番号 特願2001-287441(P2001-287441)

(22) 出願日 平成13年9月20日 (2001.9.20)

(31) 優先権主張番号 09/738568

(32) 優先日 平成12年12月15日 (2000.12.15)

(33) 優先権主張国 米国 (U S)

(31) 優先権主張番号 60/244314

(32) 優先日 平成12年10月30日 (2000.10.30)

(33) 優先権主張国 米国 (U S)

(71) 出願人 391055933

マイクロソフト コーポレーション
MICROSOFT CORPORATI
ON

アメリカ合衆国 ワシントン州 98052-
6399 レッドモンド ワン マイクロソフ
ト ウェイ (番地なし)

(72) 発明者 ニール クリフト

アメリカ合衆国 98052 ワシントン州
レッドモンド 145 コート 8201 ノー
スイースト 8201

(74) 代理人 100077481

弁理士 谷 義一 (外2名)

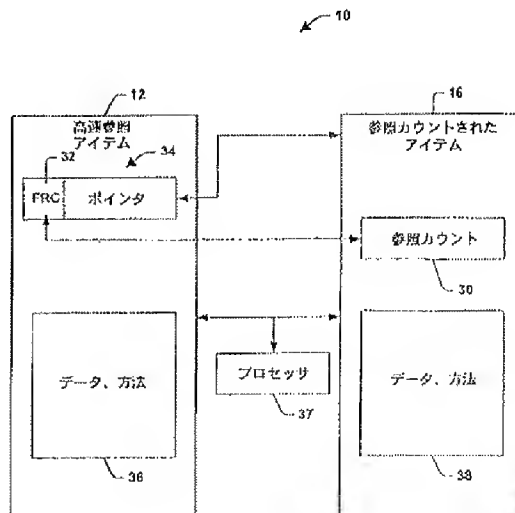
最終頁に続く

(54) 【発明の名称】 参照カウントされたアイテムを高速参照するためのシステムおよび方法

(57) 【要約】

【課題】 参照カウントされたアイテムを高速参照するためのシステムおよび方法を提供する。

【解決手段】 システムが、参照カウントされたアイテムを確立し、高速参照アイテムを確立し、参照カウントされたアイテムへのポインタを設定し、参照カウントされたアイテムのローカルコピーを、参照カウントされたアイテムへのポインタに格納することを備え、ポインタは高速参照アイテムに関連付けられる。したがって、システムおよび方法により、参照カウントされたアイテムがロック、アクセス、アンロックされるロックアルゴリズムを使用することなく、アイテムが、参照カウントされたアイテムへの参照を高速参照アイテムから得ることができる。



【特許請求の範囲】

【請求項1】 参照カウントされたアイテム（RCI）を高速参照するためのシステムであって、第1の高速参照アイテムと、高速参照カウント（FRC）と、前記FRCを確立するように適合された高速参照カウント作成器とを含み、前記FRCが前記第1の高速参照アイテムに関連付けられ、前記FRCは前記RCIに関連付けられた値Fを有し、前記値Fは最初にGより大きく、Gは、前記RCIを参照する高速参照アイテムおよび非高速参照アイテムの数を表す値であり、前記RCIを参照することを求める、1つまたは複数の第2の参照アイテムが、前記FRCを使用して前記RCIへのアクセスを容易にすることを特徴とするシステム。

【請求項2】 前記高速参照カウント作成器が前記値Fを、前記RCIへのポインタ内の1つまたは複数のビットに格納することを特徴とする、請求項1に記載のシステム。

【請求項3】 前記FRCが前記値Fに設定され、該FはXよりも小さく、該Xは、前記RCIに関連付けられた参照カウントであることを特徴とする、請求項2に記載のシステム。

【請求項4】 前記FがX-1に等しく設定されることを特徴とする請求項3に記載のシステム。

【請求項5】 前記FRCを格納する前記1つまたは複数のビットが連続していることを特徴とする、請求項2に記載のシステム。

【請求項6】 前記FRCを格納する前記1つまたは複数のビットが連続していないことを特徴とする、請求項2に記載のシステム。

【請求項7】 3つのビットを使用して前記FRCを格納することを特徴とする、請求項6に記載のシステム。

【請求項8】 前記1つまたは複数の第2の参照アイテムを処理するように適合されたプロセッサをさらに含み、前記第2の参照アイテムが、前記RCIへの1つまたは複数の第2の参照を、前記FRCを使用することによって選択的に獲得することを特徴とする、請求項1に記載のシステム。

【請求項9】 前記1つまたは複数の第2の参照アイテムが、前記RCIへの前記1つまたは複数の第2の参照を、前記第1の高速参照アイテムから、前記第1の高速参照アイテムに関連付けられた前記FRCの前記値が所定のしきい値Aよりも大きいときに、選択的に獲得し、該Aが整数であることを特徴とする、請求項8に記載のシステム。

【請求項10】 前記Aがゼロに等しいことを特徴とする請求項9に記載のシステム。

【請求項11】 前記1つまたは複数の第2の参照アイ

テムが、前記RCIへの1つまたは複数の第2の参照を解放するように動作可能であることを特徴とする、請求項9に記載のシステム。

【請求項12】 前記1つまたは複数の第2の参照アイテムが、前記RCIを直接デリファレンスすることを特徴とする、請求項11に記載のシステム。

【請求項13】 前記1つまたは複数の第2の参照アイテムが、前記1つまたは複数の第2の参照を、前記1つまたは複数の第2の参照を獲得した前記1つまたは複数の第2の参照アイテムから元の前記第1の高速参照アイテムに戻すことを特徴とする、請求項11に記載のシステム。

【請求項14】 前記第2の参照アイテムが、データ構造、オブジェクト、処理、または実行のスレッドであることを特徴とする、請求項11に記載のシステム。

【請求項15】 前記参照カウントされたアイテムが、記憶位置、データ構造、およびオブジェクトのうちの1つであることを特徴とする、請求項11に記載のシステム。

20 【請求項16】 前記第1の参照アイテムが、データ構造、オブジェクト、処理、または実行のスレッドであることを特徴とする、請求項11に記載のシステム。

【請求項17】 参照カウントされたアイテムを高速参照するための方法であって、

参照カウントされたアイテムへの第1の参照を設定するステップを含み、該第1の参照を設定するステップが参照カウントXを前記参照カウントされたアイテムに関連付けさせ、該Xは1よりも大きい整数であり、さらに、前記参照カウントされたアイテムのアドレスを、該参照カウントされたアイテムへのポインタに格納するステップと、前記参照カウントXに関連付けられた第1のローカル値を格納するステップとを含み、該第1のローカル値が前記第1の参照アイテムに関連付けられることを特徴とする方法。

【請求項18】 1つまたは複数の第2の参照アイテムが、前記参照カウントされたアイテムへの1つまたは複数の第2の参照を得るステップをさらに含み、該第2の参照を得るステップが、

40 前記参照カウントXに関連付けられた前記第1のローカル値を検査するステップと、

アトミック交換を選択的に実行するステップとを含み、前記第1のローカル値の値が1だけ減らされ、1は整数であり、第2の参照アイテムが、前記参照カウントXに関連付けられた第2のローカル値を確立し、前記第2のローカル値が前記整数値1-1に設定され、該第2のローカル値が前記第2の参照アイテムに関連付けられることを特徴とする、請求項17に記載の方法。

【請求項19】 解放アイテムが、前記参照カウントされたアイテムへの1つまたは複数の第2の参照を解放す

るステップをさらに含み、該第2の参照を解放するステップが、前記参照カウントされたアイテムをデリファレンスするステップを含むことを特徴とする、請求項18に記載の方法。

【請求項20】 前記参照カウントされたアイテムをデリファレンスするステップが、前記参照カウントXを、前記解放アイテムによって保持された前記参照の数に等しい量だけ減らすステップと、前記解放アイテムにおける前記第2のローカル値を値Jに設定するステップとを含み、該Jは整数であり、該Jは、前記解放アイテムがもはや前記参照カウントされたアイテムへの参照を有していないことを指示することを特徴とする、請求項19に記載の方法。

【請求項21】 前記Jがゼロに等しいことを特徴とする、請求項20に記載の方法。

【請求項22】 解放アイテムが、前記参照カウントされたアイテムへの1つまたは複数の第2の参照を解放するステップをさらに含み、該第2の参照を解放するステップが、アトミック交換を実行するステップを含み、前記解放アイテムが参照カウントされたオブジェクトへのその参照を獲得した元の前記アイテムに関連付けられた前記第1のローカル値の前記値が、量1だけ増分され、該1は、前記解放アイテムによって獲得された参照の数に等しい整数であり、さらに、前記解放アイテムにおける前記第2のローカル値を、事前定義された値Jに設定するステップを含み、該Jは整数であり、該Jが、前記解放アイテムがもはや前記参照カウントされたオブジェクトへの参照を有していないことを指示することを特徴とする、請求項18に記載の方法。

【請求項23】 前記アトミック交換を選択的に実行するステップが、前記第1のローカル値が所定のしきい値Sよりも大きいときに起こり、該Sは整数であることを特徴とする、請求項18に記載の方法。

【請求項24】 前記Sがゼロに等しいことを特徴とする、請求項23に記載の方法。

【請求項25】 前記第2の参照アイテムが、データ構造、オブジェクト、処理、または実行のスレッドであることを特徴とする、請求項18に記載の方法。

【請求項26】 1つまたは複数の第2の参照アイテムが、前記参照カウントされたアイテムへの1つまたは複数の第2の参照を得るステップをさらに含み、該第2の参照を得るステップが、前記第1のローカル値を検査するステップと、第2の参照を、第1の参照を得るための方法を使用して、選択的に得るステップとを含むことを特徴とする、請求項17に記載の方法。

【請求項27】 前記ポインタにおける1つまたは複数のビットを使用して前記第1のローカル値を格納し、該

ビットが連続していることを特徴とする、請求項17に記載の方法。

【請求項28】 3つのビットを使用して前記第1のローカル値を格納することを特徴とする、請求項17に記載の方法。

【請求項29】 前記ポインタにおける1つまたは複数のビットを使用して前記第1のローカル値を格納し、該ビットが連続していないことを特徴とする、請求項17に記載の方法。

10 【請求項30】 前記第1のローカル値が値Yに設定され、該Yは前記参照カウントXよりも小さい整数であることを特徴とする、請求項17に記載の方法。

【請求項31】 前記Yが前記参照カウントXよりも1だけ小さく設定されることを特徴とする、請求項30に記載の方法。

【請求項32】 前記参照カウントされたアイテムが、記憶位置、データ構造、またはオブジェクトであることを特徴とする、請求項17に記載の方法。

20 【請求項33】 前記第1の参照アイテムが、データ構造、オブジェクト、処理、または実行のスレッドであることを特徴とする、請求項17に記載の方法。

【請求項34】 請求項17に記載の方法を実行するように動作可能であるコンピュータ実行可能命令を有することを特徴とする、コンピュータ可読媒体。

【請求項35】 第1のコンピュータシステムから第2のコンピュータシステムへ伝送されるように適合されたデータパケットであって、該データパケットは参照カウントされたオブジェクトへの参照を獲得することに関する情報を含むことを特徴とするデータパケット。

30 【請求項36】 前記参照カウントされたオブジェクトへのポインタに格納された参照カウントに関連付けられた、第1のローカル値に関する情報をさらに含むことを特徴とする、請求項35に記載のデータパケット。

【請求項37】 その上にデータ構造を格納しているコンピュータ可読媒体であって、

参照カウントされたアイテムへのポインタに関する第1のデータを含む第1のデータフィールドを含み、該ポインタは、該参照カウントされたアイテムに関連付けられた参照カウントに関連付けられた第1のローカル値を格納するように動作可能な1つまたは複数のビットを含むことを特徴とする、コンピュータ可読媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、一般にコンピュータプログラミングに関する。より詳細には、本発明は、参照カウントされたアイテムの高速参照に関する。

【0002】

【従来の技術】参照カウントは、メモリ管理システムにおいて使用される。参照カウントされたアイテムは、それに対して存在する参照された数のカウントを保つ。参

照が存在しないアイテムは、使用可能な空きメモリとして回収することができる。参照カウントは典型的には1のカウントで開始され、各参照が追加されるごとに1ずつ増分され、各参照が解消されるごとに1ずつ減分される。このようなシステムでは、参照カウントがゼロになったとき、このアイテムはもはや必要とされず、再使用することができ、したがって関連付けられたメモリが解放される。正確な参照カウントを維持することは、メモリ管理に重要である。たとえば、参照カウントが、アイテムへの参照がないことを示し、参照カウントされたアイテムを参照するアプリケーションがあるが、参照カウントされたアイテムが削除された場合には、そのアプリケーションが「未解決なポインタ」の問題を経験する可能性があり、あるいは、そのアプリケーションが、もはやそのアプリケーションに関連付けられたアイテムを表さないメモリを読み込まれている可能性がある。さらなる例として、参照カウントが、アイテムへの参照があったことを示しているが、このアイテムを参照するエンティティ（実体）がなかった場合には、このアイテムは再利用されず、参照カウントされたアイテムが関連付けられたシステムが「メモリリーク」の問題を被る可能性がある。したがって、参照カウントが正確に維持されることを保証することが重要である。したがって、参照カウントが頻繁にロックによって保護され、これにより、一度にただ1つのエンティティのみが参照カウントにアクセスすることを容易にすることで、未解決なポインタおよびメモリリークの問題が軽減される。しかし、ロック機構は、たとえば、2つ以上のエンティティがロックを同時点に獲得することを求めたとき、競合の問題を生じる。ロックを獲得したエンティティ以外のエンティティは待機しなければならず、これによりシステム性能に否定的な影響を与える。

【0003】オペレーティングシステム、マルチスレッド（多重系）アプリケーション、およびサーバは、たとえば、しばしば、変更可能なサブデータに関連付けられたデータ構造を維持する。たとえば、コンピュータシステムのユーザがプロファイルを有する可能性があるが、これは1組のアクセス権（たとえば、ファイルを読むことができるが、ファイルに書き込むことができない）を含む。サブデータ（たとえば、アクセス権）にアクセスするには、従来の参照カウント方法では、サブデータへのポインタを保護するロックを獲得し、サブデータへのポインタを取り出し、サブデータに関連付けられた参照カウントを増分し、次いでロックを解除する。したがって、ロックおよびアンロック動作がサブデータの各アクセスについて使用され、これにより、このようなオペレーティングシステム、マルチスレッドアプリケーションおよびサーバの処理速度が、たとえば上に記載した競合の問題により低下する。サブデータにアクセスする並行プロセッサの数が増加するにつれて、従来のロック、取

り出し、増分、ロック解除方法の性能は急速に低下するが、これは、より多くのエンティティがロックを求めて競合するからである。このような従来方法は特に、サブデータが頻繁に読み出されるがめったに変更されないとき、システムリソース（たとえば、CPUサイクル）を無駄にする。

【0004】

【発明が解決しようとする課題】したがって、参照カウントされたアイテムを高速参照するためのシステムおよび方法で、方法の性能がそれほど急速に低下しないものが必要とされる。

【0005】以下に、本発明のいくつかの態様についての基本的な理解を提供するために、本発明の簡単な要約を提示する。この要約は、本発明の広範な概要ではない。本発明のキーとなる、あるいは臨界の要素を明らかにすることも、本発明の権利範囲を線引きすることも意図されるものではない。この要約の唯一の意図は、本発明のいくつかの概念を簡単な表現形式において、後に提示するより本発明の詳細な説明の序文として提示することにある。

【0006】本発明は、参照カウントされたアイテムを高速参照して、並行プロセッサを使用するマルチスレッドアプリケーションおよび/またはシステムの性能を助長するためのシステムおよび方法を提供する。

【0007】

【課題を解決するための手段】本発明では、高速参照アイテムが、参照カウントされたアイテムにおける参照カウントを1より大きい値に確立し、参照カウントされたアイテムへのポインタを確立し、参照カウントに関連付けられたローカル値を格納するが、これは高速参照カウント（FRC）として知られている。最初のFRC値は、参照カウントされたアイテムを参照する参照アイテムの数より大きい。

【0008】参照カウントされたアイテムにおける参照の数を1より大きく確立することによって、参照アイテムが、複数の参照を獲得したとみなすことができる。次いで、「エキストラ（臨時の）」参照が、参照カウントされたアイテムへの参照を求める後続のアイテムで使用可能であり、一定の条件下で、このような後続の獲得アイテムが、ロックを獲得し、参照カウントを変更し、ロックを解除することを要求されない。むしろ、後続の獲得アイテムが、高速参照アイテムにおいて使用可能な参照の数を検査し、1つまたは複数の「エキストラ」参照を獲得（たとえば、「盗む／借りる」）ことができる。この方法を使用して参照を獲得するとき、アトミックオペレーション（たとえば、InterlockedCompareExchange（））を使用して、高速参照アイテムにおけるFRCを変更して、参照カウントされたアイテムへの参照の正確なカウントの維持を容易にする。複数の参照を獲得する高速参照アイテムは、それ

自体で使用可能な参照について検査することができ、その獲得された使用可能な参照の1つまたは複数を含むことができる。

【0009】FRCを、ポインタにおける可変の、あるいは、1つまたは複数のビットに格納することができる。このようなビットは、たとえば、参照カウントされたアイテムがメモリ領域に位置し、アイテムが8バイト境界に合わせられた（たとえば、ポインタの下位3ビットを解放する）場合、あるいは、ポインタにおける他のビットが使用可能であった場合、使用可能にすることができる。FRCを格納するために使用されたビットは、ポインタにおいて連続にすることができ、あるいは、これらをポインタ中で分散させることができる。

【0010】したがって、このシステムおよび方法は、ロックを獲得し、参照を獲得し、ロックを解除するロックアルゴリズムを使用することなく、参照カウントされたアイテムへの参照を得ることを容易にする。したがって、多数のエンティティが、参照カウントされたアイテムへのロックを獲得することを求めているときに生じる競合の問題による、性能低下が緩和される。

【0011】

【発明の実施の形態】前述および関連する本発明の目的を達成するための本発明の実施についての、本発明の例示的な実施形態を、添付の図面に基づき以下に詳述する。ただ、これらの実施形態は、本発明の原理を使用することができる様々な方法のほんの数例を示したものにすぎず、本発明は、このようなすべての形態およびそれらの均等物を含むことを意図するものである。本発明の他の効果および新規な特徴は、以下の本発明の詳細な説明を、図面と共に考察したときに明らかになるであろう。

【0012】さて、本発明を、図面を参照して説明し、類似の参照番号を使用して全体で類似の要素を参照する。以下の記述では、本発明の完全な理解を提供するために多数の特定の細部を述べる。しかし、本発明をこれらの特定の細部なしに実施できることは、当業者には明らかであろう。別の例では、よく知られた構造およびデバイス、本発明の記載を容易にするために、ブロック図の形式において図示する。

【0013】最初に図1を参照して、参照カウントされたアイテム16を高速参照するためのシステム10を例示する。高速参照アイテム12および非高速参照アイテム14を例示する。参照カウントされたアイテム16は、たとえば、メモリ場所、データ構造、および/またはオブジェクトにすることができる。参照カウントされたアイテム16は、たとえば、コンピュータユーザプロフィールに関連して使用することができる。プロフィールは、不変データおよび可変データを含む可能性がある。可変データは、たとえば、ファイルの読み取り、書き込みおよび実行に係る、コンピュータユーザの資

格証明書(credential)にすることができる。可変データを、参照カウントされたアイテム16における参照カウントに関連付けることができる。可変データは変更することができるが、変更の可能性が低かった場合、可変データへの参照のカウントに関連付けられたローカル値が、高速参照アイテム12におけるキャッシング(caching)に適している。したがって、可変データへの参照のカウントに関連付けられたローカル値が、他の高速参照アイテム12および非高速参照アイテム14で使用可能である。

【0014】ここで図2を参照して高速参照アイテム12を例示するが、これは高速参照カウント33(FRC)を含み、これがF参照を表す値を保持し、これらが、参照カウントされたアイテム16における参照カウント31に格納された参照のカウントXに関係する。FRC33を、高速参照カウント作成器35によって作成することができる。たとえば、オペレーティングシステム参照カウントマネージャが、高速参照カウント作成器35として機能し、高速参照カウント33を作成することができる。高速参照カウント作成器35は、コンピュータに関連したエンティティであって、ハードウェア、ハードウェアおよびソフトウェアの組み合わせ、ソフトウェア、または実行中のソフトウェアとすることができる。たとえば、高速参照カウント作成器35を、プロセッサ上で動作中の処理、オペレーティングシステムコンポーネント(構成要素)、プロセッサ、オブジェクト、実行可能なもの、実行スレッド(thread)、プログラムおよびコンピュータにすることができる。Fの値は最初に、参照カウントされたアイテム16を参照するアイテムの実際の数より大きい値に設定され、これにより、後続の参照アイテムが高速参照アイテム12から参照を獲得することを容易にする。図2は、新しい参照アイテム15も例示し、これは、参照カウントされたアイテム16への参照を、高速参照アイテム12から、FRC33を使用することによって獲得しようと試みることができる。したがって、参照カウントされたアイテム16に関連付けられたロックを獲得かつ解除することに関係付けられた問題を緩和する。たとえば、新しい参照アイテム15が、FRC33からのカウント値を、参照カウントされたアイテム16を参照しようとするその試みにおいて利用することができる。したがって、他の参照アイテムによって使用することができるエキストラ参照カウントを有するFRC33によって、FRC33が、他の参照アイテムによる、参照カウントされたアイテム16の高速参照を容易にする。

【0015】ここで、図3を参照して、システム10をより詳細に例示する。参照カウントされたアイテム16が参照カウント30を含み、これは、参照カウントされたアイテム16を参照中であるアイテムの数のカウントである。参照カウントされたアイテム16は、他の情

報、たとえば、データおよび/または方法38を含むこともできる。高速参照アイテム12は、参照カウントされたアイテム16へのポインタ34を維持し、参照カウント30に関連付けられたローカル値（これをポインタの一部にすることができる）も維持し、これは、本明細書では高速参照カウント32（FRC）と称するものとする。本発明の一形態によれば、参照カウントされたアイテム16を作成するとき、高速参照アイテム12が、参照カウントされたアイテム16へのロックを獲得し、参照カウント30を1より大きい値に確立し、ロックを解除する。高速参照アイテム12が、参照カウントされたアイテム16を参照するただ1つのアイテムであっても、参照カウント30を、参照カウントされたアイテム16を参照するアイテムの実際の数より大きい数で確立することによって、高速参照アイテム12によって獲得されたエキストラ参照を、ある条件下で、他の高速参照アイテム12および非高速参照アイテム14（図1）によって、ロックを獲得かつ解除する要件なしに獲得することができる。このような条件は、高速参照アイテム12が、参照カウント30に関連付けられたサブデータにおける値を読み取るが変更しないことを求めたとき、存在する。もう1つのこのような条件は、非高速参照アイテム14が、参照カウント30に関連付けられたサブデータにおける値を読み取るが、変更しないことを求めたとき、存在する。高速参照アイテム12または非高速参照アイテム14のいずれかが、参照カウント30に関連付けられたサブデータを変更することを求めた場合、これらのアイテムが従来のロック獲得、アクセス、ロック解除方法を使用する。

【0016】しかし、後続の獲得アイテムが、参照カウント30に関連付けられたデータの読み取りを可能にするための参照を獲得することを求めたとき、次いで、獲得アイテムがプロセッサ37を使用して、高速参照アイテム12におけるFRC32を検査して、いずれかの参照が、獲得アイテムによって獲得されるために使用可能であるかどうかを決定することができる。1つまたは複数の参照が、獲得されるために使用可能であった場合、獲得アイテムがアトミックオペレーション（たとえば、Interlocked Compare Exchange ()）をプロセッサ37上で実行させて、FRC32を、獲得アイテムによって獲得された参照の数だけ減らし、それ自体のローカル参照カウントを確立する。FRC32を検査し、アトミック更新動作を実行し、ローカル参照カウントを獲得アイテムにおいて確立することなく、プロセッサ37がロックを獲得かつ解除することなく、実行される。したがって、参照カウントされたアイテム16への参照を獲得するための従来の方法に関連付けられた競合の問題が緩和される。獲得アイテムが高速参照アイテム12であった場合、獲得アイテムはFRC32を、他の後続の獲得者によって検査かつ操作させる

ことができる。プロセッサ37は、コンピュータに関連したエンティティであり、ハードウェア、ハードウェアおよびソフトウェアの組み合わせ、ソフトウェア、または実行中のソフトウェアにすることができる。たとえば、プロセッサを、マイクロプロセッサ上で動作中の処理、マイクロプロセッサ、オブジェクト、実行可能なもの、実行スレッド、プログラムおよびコンピュータにすることができる。例示のため、サーバ上で動作中のアプリケーションおよびサーバをプロセッサにすることができる。

【0017】ここで、図4を参照して、高速参照カウント52（FRC）を格納するための2つの具体例としての方法を示す。第1の方法では、Nビットからなるポインタ40、および、Xビットからなる変数42が使用される。ポインタ40は、参照カウントされたアイテム16（図1）のアドレスを保持し、変数42は高速参照カウント52（FRC）を保持する。第2の方法では、Nビットからなるポインタ50が、Mビットからなるポインタ部分、および、N-MビットからなるFRC52部分の2つの部分に区分される。たとえば、ポインタ50が32ビットからなることができ、そのうちの29ビットが、参照カウントされたアイテム16（図1）のアドレスの格納に割り付けられ、3ビットがFRC52を格納するために割り付けられる。Nビットのすべてが、参照カウントされたアイテム16（図1）のアドレスを格納するために必要とされるのではないとき、ポインタ50のビットを使用してFRC52を格納することができる。たとえば、参照カウントされたアイテム16がメモリ領域に位置し、データアイテムが8バイト境界に合わせられた場合、32ビットのうちの3ビットは、アドレスを格納するために必要とされない。したがって、この使用可能な3ビットを使用してFRC52を格納することができ、値はゼロから7の範囲となる。図4では、FRC52を格納するために使用された3ビットを互いに連続して例示するが、以下で図5に関連して記載するように、これらのビットが連続している必要はない。32ビットおよび3ビットを上で論じたが、本発明によれば、より多い数あるいは少ない数のビットを使用できることを、当業者には理解されたい。

【0018】第1の方法に勝る第2の方法の利点は、ポインタ50において使用可能なビットを使用することによって、メモリ要件を低減させることができ、処理速度を向上させることができることである。たとえば、第1の方法における2つのメモリルックアップに対して、1つのメモリルックアップを使用して、ポインタ50におけるアドレスを解決し、FRC52を検索することができる。いずれの場合も、後続の獲得アイテムによってロックを獲得することなく、FRC52を検査して、参照が獲得されるために使用可能であるかどうかを決定することができる。参照が使用可能であり、後続の獲得アイ

テムが、参照カウント30(図3)に関連付けられたサブデータを読み取るが変更しないことを求めた場合、後続の獲得アイテムが、ロックを獲得かつ解除せずに、アトミックオペレーションを実行してFRC52を変更することができる。したがって、参照カウントされたアイテムへの参照を獲得するためにロックを求める競合に関連付けられた問題が緩和される。

【0019】図5は、高速参照カウント60(FRC)を格納するように適合されたポイント68を例示し、FRC60を格納するために使用されたビットが連続していない。ポイント68はNビットを有することができる。参照カウントされたアイテム16(図1)のアドレスを格納するためにMビットが必要とされた場合、N-Mビットが、FRC60を格納するために使用可能である。図5では、N-Mビット、FRC6162、FRC6264ないしFRC6466が、ポイント68中で分散される。N-Mビットを、ポイント68において左から右へ配置して例示するが、N-Mビットのいかなる適した配置も本発明において使用できることを、当業者には理解されたい。

【0020】図6は、タイミング図であり、参照カウント90が確立され、操作されるとき、参照カウントされたアイテム72における参照カウント90を例示する。高速参照アイテム70が、参照カウントされたアイテム72を確立する。参照カウントされたアイテム72の確立において、高速参照アイテム70が参照カウント90および高速参照カウント80を確立する。参照カウント90、高速参照カウント80、および高速参照カウント100を、4つの異なる時点T1、T2、T3およびT4で例示する。T1は、高速参照アイテム70が作成され、参照カウントされたアイテム72を作成する時点に対応する。高速参照アイテム70が参照カウント90A1を値8で確立し、8つのアイテムが、参照カウントされたアイテム72を参照中であることを指示する。しかし、時点T1で、ただ1つのアイテムのみが、参照カウントされたアイテム72を参照中である。したがって、参照カウントされたアイテム72への7つの参照が、参照カウントされたオブジェクト72に関連付けられたサブデータを読み取るが変更しないことを求める他のアイテムによって、獲得されるために使用可能である。時点T1で、第2の高速参照アイテム74はまだ作成されておらず、したがって、高速参照カウント100A1のための値が定義されていない。時点T2で、高速参照アイテム74が作成される。高速参照アイテム74は、参照カウントされたアイテム72に関連付けられたサブデータを読み取るが変更しないことを求め、したがって、高速参照アイテム74が、参照を高速参照アイテム70から獲得しようと試みることができる。高速参照アイテム74がFRC80A1を検査し、参照が使用可能であることを決定し、FRC80A1のアトミック更新に、FRC8

0A1を、高速参照アイテム74が獲得中である1つの参照だけ減らすようにさせる。アトミックオペレーションは、同時に獲得中の他のアイテムがFRC80A1を、高速参照アイテム74と同時に検査かつ変更しないようにすることを容易にするために、必要とされる。アトミックオペレーションが完了したとき、時点T2で、高速参照アイテム70がFRC80A2を6に減らし、6つの参照が獲得されるために使用可能であることを指示し、高速参照アイテム74がFRC100A2をゼロに設定し、これが、獲得されるために使用可能な参照を有していないことを指示する。参照カウントされたオブジェクト72における参照カウント90A2は変更されず、これは、1つの参照が高速参照アイテム70によって使用中であり、1つの参照が高速参照アイテム74によって使用中であり、6つの参照が、高速参照アイテム70から使用可能であるとして列挙され、これらが合計で、参照カウント90A2において指示された8つの参照となるからである。代替例では、高速参照アイテム74が複数の参照を高速参照アイテム70から獲得することができ、この場合、高速参照カウント80A2が、高速参照アイテム74によって獲得された参照の数だけ減らされ、高速参照カウント100A2が、獲得された参照の数より1だけ小さく設定され、高速参照アイテム74から獲得されるために使用可能な参照の数を指示する。このような代替獲得の後、高速参照アイテム70および高速参照アイテム74が、いずれかがなお他のアイテムが獲得するために適切な参照を有しているかどうかを決定するために、検査されるために使用可能である。

【0021】時点T3で、高速参照アイテム70が、参照カウントされたアイテム72への関心(interest)を失い、したがって、参照カウント90A3を1に変更し、6つの「エキストラ」参照がもはや使用可能でないこと、および、高速参照アイテム70によって消費された1つの参照がもはや使用中でないことを指示する。高速参照アイテム70がFRC80A3をヌル値に設定し、高速参照アイテム70がもはや参照カウントされたアイテム72を参照中でないことを指示する。参照カウント90A3における値1は、正確に、1つのアイテム、高速参照アイテム74が、参照カウントされたアイテム72を参照中であることを指示する。

【0022】時点T4で、高速参照アイテム74が、参照カウントされたアイテム72への関心を失い、したがって、参照カウント90A4をゼロに変更し、参照カウントされたオブジェクト72を参照中のアイテムがないことを指示する。参照カウント90A4におけるゼロは、参照カウントされたアイテム72に関連付けられたメモリが、たとえば、ガベージコレクション処理によって再生利用できることを指示する。高速参照アイテム74がFRC100A4をヌル値に設定し、高速参照アイテム74がもはや参照カウントされたアイテム72を参照中では

いことを指示する。

【0023】高速参照アイテム70が最初に、参照カウントされたアイテム72において8のカウントを確立したが、1より大きいいかなる適切な値も、参照カウントされたアイテム72において確立することができ、後続の獲得アイテムがロックの獲得かつ解除をせずに、参照カウントされたアイテム72への参照を獲得することを容易することができることを、当業者には理解されたい。したがって、ロックを求める競合に関連付けられた問題が緩和される。高速参照アイテム74が複数の参照を高速参照アイテム70から獲得できることを、さらに理解されたい。

【0024】図7はタイミング図であり、参照カウント130が確立され、操作されるとき、参照カウントされたアイテム112における参照カウント130を例示する。高速参照アイテム114が、参照カウントされたアイテム112を確立する。参照カウントされたアイテム112の確立において、高速参照アイテム114が参照カウント130および高速参照カウント120を確立する。参照カウント130、高速参照カウント120、および非高速参照カウント140を、4つの異なる時点T₁、T₂、T₃およびT₄で例示する。T₁は、高速参照アイテム114が作成され、参照カウントされたアイテム112を作成する時点に対応する。高速参照アイテム114が参照カウント130A₁を値8で確立し、8つのアイテムが、参照カウントされたアイテム112を参照中であることを指示する。しかし、時点T₁で、ただ1つのアイテムのみが、参照カウントされたアイテム112を参照中である。したがって、参照カウントされたアイテム112への7つの参照が、参照カウントされたオブジェクト112に関連付けられたサブデータを読み取るが変更しないことを求める他のアイテムによって、獲得されるために使用可能である。時点T₁で、非高速参照アイテム116はまだ作成されておらず、したがって、非高速参照カウント140A₁のための値が定義されていない。

【0025】時点T₂で、非高速参照アイテム116が作成される。非高速参照アイテム116は、参照カウントされたアイテム112に関連付けられたサブデータを読み取るが変更しないことを求め、したがって、非高速参照アイテム116が、参照を高速参照アイテム114から獲得しようと試みる。非高速参照アイテム116がFRC120A₁を検査し、参照が使用可能であることを決定し、FRC120A₁のアトミック更新に、FRC120A₁を、非高速参照アイテム116が獲得中である1つの参照だけ減らすようにさせる。アトミックオペレーションは、同時に獲得中の他のアイテムがFRC120A₁を、非高速参照アイテム116と同時に検査かつ変更しないようにすることを容易にするために、必要とされる。アトミックオペレーションが完了したとき、時点T

2で、高速参照アイテム114がFRC120A₂を6に減らし、6つの参照が獲得されるために使用可能であることを指示し、非高速参照アイテム116がFRC140A₂を1に設定し、これが参照を消費中であることを指示する。非高速参照アイテム116を使用して参照をそこから獲得することはできず、したがって、その非高速参照カウント140は、参照が使用可能であることを指示せず、むしろそれが参照を消費中であることを指示する。参照カウントされたオブジェクト112における参照カウント130A₂は変更されず、これは、1つの参照が高速参照アイテム114によって使用中であり、1つの参照が非高速参照アイテム116によって使用中であり、6つの参照が、高速参照アイテム114から使用可能であるとして列挙され、これらが合計で、参照カウント130A₂において指示された8つの参照となるからである。

【0026】時点T₃で、高速参照アイテム114が、参照カウントされたアイテム112への関心を失い、したがって、参照カウント130A₃を1に変更し、6つの「エキストラ」参照がもはや使用可能でないこと、および、高速参照アイテム114によって消費された1つの参照がもはや使用中でないことを指示する。高速参照アイテム114がFRC120A₃をヌル値に設定し、高速参照アイテム114がもはや参照カウントされたアイテム112を参照中でないことを指示する。参照カウント130A₃における値1は、正確に、1つのアイテム、非高速参照アイテム116が、参照カウントされたアイテム112を参照中であることを指示する。

【0027】時点T₄で、非高速参照アイテム116が、参照カウントされたアイテム112への関心を失い、したがって、参照カウント130A₄をゼロに変更し、参照カウントされたオブジェクト112を参照中のアイテムがないことを指示する。参照カウント130A₄におけるゼロは、参照カウントされたアイテム112に関連付けられたメモリが、たとえば、ガベージコレクション処理によって再生利用できることを指示する。非高速参照アイテム116がFRC140A₄をヌル値に設定し、非高速参照アイテム116がもはや参照カウントされたアイテム112を参照中でないことを指示する。

【0028】高速参照アイテム114が最初に、参照カウントされたアイテム112において8のカウントを確立したが、1より大きいいかなる適切な値も、参照カウントされたアイテム112において確立することができ、後続の獲得アイテムがロックの獲得かつ解除をせずに、参照カウントされたアイテム112への参照を獲得することを容易することができることを、当業者には理解されたい。したがって、ロックを求める競合に関連付けられた問題が緩和される。

【0029】図8はタイミング図であり、参照カウント170が確立され、操作されるとき、参照カウントさ

れたアイテム152における参照カウント170を例示する。高速参照アイテム150が、参照カウントされたアイテム152を確立する。参照カウントされたアイテム152の確立において、高速参照アイテム150が参照カウント170および高速参照カウント160を確立する。参照カウント170、高速参照カウント160、および非高速参照カウント180を、4つの異なる時点T₁、T₂、T₃およびT₄で例示する。T₁は、高速参照アイテム150が作成され、参照カウントされたアイテム152を作成する時点に対応する。高速参照アイテム150が参照カウント170A₁を値8で確立し、8つのアイテムが、参照カウントされたアイテム152を参照中であることを指示する。しかし、時点T₁で、ただ1つのアイテムのみが、参照カウントされたアイテム152を参照中である。したがって、参照カウントされたアイテム152への7つの参照が、参照カウントされたオブジェクト152に関連付けられたサブデータを読み取るが変更しないことを求める他のアイテムによって、獲得されるために使用可能である。時点T₂で、非高速参照アイテム154はまだ作成されておらず、したがって、非高速参照カウント160A₁のための値が定義されていない。時点T₂で、非高速参照アイテム154が作成される。非高速参照アイテム154は、参照カウントされたアイテム152に関連付けられたサブデータを読み取るが変更しないことを求め、したがって、非高速参照アイテム154が、参照を高速参照アイテム150から獲得しようと試みることができる。非高速参照アイテム154がFRC160A₁を検査し、参照が使用可能であることを決定し、FRC160A₁のアトミック更新に、FRC160A₁を、非高速参照アイテム154が獲得中である1つの参照だけ減らすようにさせる。アトミックオペレーションは、同時に獲得中の他のアイテムがFRC160A₁を、非高速参照アイテム154と同時に検査かつ変更しないようにすることを容易にするために、必要とされる。アトミックオペレーションが完了したとき、時点T₂で、高速参照アイテム150がFRC160A₂を6に減らし、6つの参照が獲得されるために使用可能であることを指示し、非高速参照アイテム154がその非高速参照カウント180A₂を1に設定し、これが参照を消費中であることを指示する。非高速参照アイテム154を使用して参照をそこから獲得することはできず、したがって、その非高速参照カウント180は、参照が使用可能であることを指示せず、むしろそれが参照を消費中であることを指示する。参照カウントされたオブジェクト152における参照カウント170A₂は変更されず、これは、1つの参照が高速参照アイテム150によって使用中であり、1つの参照が非高速参照アイテム154によって使用中であり、6つの参照が、高速参照アイテム150から使用可能であるとして列挙され、これらが合計で、参照カウント170A₂において

指示された8つの参照となるからである。

【0030】少なくとも2つの方法を、非高速参照アイテム154が獲得した参照を、参照カウントされたアイテム152における関心を失ったときに解放するため、使用可能である。第1の方法では、非高速参照オブジェクト154が、参照カウントされたアイテム152を直接デリファレンスする。したがって、時点T₃で、非高速参照アイテム154が、参照カウントされたアイテム152への関心を失い、参照カウント170A₃を7に変更し、非高速参照アイテム154によって消費された1つの参照がもはや使用中でないことを指示する。非高速参照アイテム154がFRC180A₃をヌル値に設定し、非高速参照アイテム154がもはや参照カウントされたアイテム152を参照中でないことを指示する。参照カウント170A₃における値7は、正確に、1つのアイテム、高速参照アイテム150が、参照カウントされたアイテム152を参照中であること、および、これが他のアイテムによる獲得のためになお使用可能である6つの「エキストラ」参照を有することを指示する。

20 【0031】もう1つの方法では、非高速参照アイテム154が、それが高速参照アイテム150から獲得した参照を戻すことができる。したがって、時点T₃で、非高速参照オブジェクトがアトミックオペレーションを実行して、高速参照カウント160A₄を7に増加し、これが、非高速参照アイテム154によって獲得された参照を戻したこと、および、7つの参照が再度獲得のために使用可能であることを指示する。

30 【0032】類似の方法で、少なくとも2つの方法を、高速参照アイテムが、参照カウントされたアイテム152をデリファレンスするために、使用可能である。第1の方法では、高速参照アイテムが高速参照カウント170を、高速参照アイテムによって解放された参照の数だけ減らすことができる。第2の方法では、高速参照アイテムが、別の高速参照アイテムから獲得された参照の数を戻し、参照カウント170を不変のままにすることができる。獲得された参照を、それらの獲得元の高速参照アイテムへ戻す前に、復帰アイテムが、参照の獲得元の高速参照アイテムがなお存在し、戻された参照を有するために使用可能であることを決定する。

40 【0033】高速参照アイテム150が最初に、参照カウントされたアイテム152において8のカウンタを確立したが、いかなる適切な値も、参照カウントされたアイテム152において確立することができ、後続の獲得アイテムがロックの獲得かつ解除をせずに、参照カウントされたアイテム152への参照を獲得することを容易にすることができることを、当業者には理解されたい。したがって、ロックを求める競争に関連付けられた問題が緩和される。

50 【0034】ここで、図9を参照して、参照カウントされたアイテム202を例示する別のタイミング図を提供

する。時点T₁で、高速参照アイテム200が作成され、参照カウントされたアイテム202を確立する。高速参照アイテム200が参照カウント220A₁を8に設定し、高速参照アイテム200が、参照カウントされたアイテム202への8つの参照を獲得したことを指示する。時点T₁で、高速参照アイテム204はまだ作成されておらず、したがって、高速参照カウント230A₁のための値がない。時点T₁で、高速参照アイテム200が高速参照カウント223A₁を7に設定し、これが、獲得されるために使用可能な7つの参照を有することを指示する。

【0035】時点T₂は、時点T₁よりも後のある期間にすることができ（たとえば、多数の獲得および/または解放が起こっている可能性がある）、ここで高速参照アイテム204が作成される。高速参照アイテム204が高速参照カウント223を検査し、参照を高速参照アイテム200から獲得できることを決定する。アトミックオペレーションを実行して高速参照カウント223を減らすことにおいて、高速参照オブジェクト204が、高速参照カウント223A₂をゼロにしたことを決定する。したがって、高速参照オブジェクト204が追加処理を実行して、参照カウントされたアイテム202への追加の参照を獲得することができかどうかを決定する。さらに参照を獲得することができた場合、高速参照アイテム204がこれらの参照を獲得し、これらをそれ自体のために保つか、あるいは、それらを高速参照アイテム200に割り当てることができる。図9では、高速参照アイテム204が、参照カウントされたアイテム202で参照が使用可能であることを、参照カウント220A₂を検査することによって決定する。参照カウント220A₂が、参照カウントされたアイテム202へ許可された参照の最大数より小さいので、高速参照アイテム204がロックを獲得し、参照カウント220A₃をT₃で8に変更し、追加の参照を高速参照アイテム200に割り当て、高速参照カウント223A₃が6に増加される。時点T₄で、高速参照アイテム204が、参照カウントされたアイテム202への関心を失い、したがって、その参照を高速参照アイテム200へ戻し、高速参照カウント223A₄を7に増加する。

【0036】高速参照アイテム204が、参照カウントされたアイテム202への6つの追加の参照を獲得したが、本発明によれば、より多い数あるいは少ない数の参照を高速参照アイテム204によって獲得できたことを、当業者には理解されたい。同様に、高速参照アイテム204が追加で獲得された参照の6つを高速参照アイテム200に割り当てたが、より少ない数を割り当てることができ、より多い数を保持できることを理解されたい。

【0037】このとき図10を参照して、本発明の一実施形態中のサンプルデータフローを例示する。高速参照

アイテム300が、参照カウントされたアイテム302を作成する。参照カウントされたアイテム302が、それに関連付けられた参照カウント308を有する。参照カウント304のコピーが、高速参照アイテム300に渡される。高速参照アイテム300が、参照カウント304に関連付けられたローカル値（たとえば、高速参照カウント）を作成し、高速参照データストア306に格納する。高速参照カウントデータストア306は、参照カウントデータストア308に格納された値とは異なる値を含むことができる。他の高速参照アイテム310が、参照カウントされたアイテム302への参照を獲得することを求めることができる。

【0038】本発明の一実施形態によれば、高速参照アイテム310が、参照カウント306データストアに格納された高速参照カウントを検査する。高速参照アイテム310が、参照が獲得されるために使用可能であると決定した場合、高速参照アイテム310がアトミックオペレーション（たとえば、InterlockedCompareExchange()）を使用して、高速参照カウント306データストアにおける使用可能な参照の数を、高速参照アイテム310によって獲得された参照の数に等しい量だけ減らすことができる。高速参照アイテム310が、獲得された参照の数に関連付けられた値を高速参照カウント314データストアに格納する。高速参照アイテム310を、参照の利用可能性について順番に検査することができ、これがそれらの「エキストラ」参照を獲得させることができる。

【0039】高速参照アイテム310が、獲得された参照により処理されたとき、高速参照アイテム310がアトミックオペレーションを使用して、高速参照カウント306データストアにおける使用可能な参照の数を、高速参照アイテム310によって解放された参照の数に等しい量だけ増加することができる。高速参照アイテム310はロックを獲得し、参照カウントされたオブジェクト302への獲得された参照を、参照カウント308データストアに格納された参照カウントを減らすことによって戻し、ロックを解除することもできる。

【0040】高速参照アイテム310は、1つまたは複数の参照を高速参照アイテム300から獲得することによって、高速参照カウント306データストアに格納された使用可能な参照が使い尽くされたことを決定することもできる。したがって、高速参照アイテム310が、参照カウントされたアイテム302への追加の参照を獲得しようと試み、追加の参照をそれら自体のために保持するか、あるいは追加の参照を高速参照アイテム300に割り当てることができる。

【0041】非高速参照アイテム320も、参照カウントされたアイテム302への参照を獲得することを求めることができる。本発明の一実施形態によれば、非高速参照アイテム320が、高速参照カウント306データ

ストアに格納されたカウントを検査する。非高速参照アイテム320が、参照が獲得されるために使用可能であると決定した場合、非高速参照アイテム320がアトミックオペレーション（たとえば、InterlockedCompareExchange()）を使用して、高速参照カウント306データストアにおける使用可能な参照の数を、非高速参照アイテム320によって獲得された参照の数に等しい量だけ減らすことができる。非高速参照アイテム320は、参照カウントされたアイテム302を直接デリファレンスするか、あるいは、獲得された参照を高速参照アイテム300へ戻すこともできる。

【0042】1つの獲得高速参照アイテム310および1つの獲得非高速参照アイテム320を例示したが、本発明によれば、より多い数あるいは少ない数の高速参照アイテムおよび／または非高速参照アイテムを使用できることを、当業者には理解されたい。

【0043】図1ないし図9に関連した考察を通じて、「アイテム」という用語を使用した（たとえば、参照カウントされたアイテム、高速参照アイテム、非高速参照アイテム）。そのように使用されたような「アイテム」という用語の意味には、メモリ場所、データ構造、オブジェクト、処理および／または実行のスレッドが含まれる可能性があるが、それらに限定されないことを、当業者には理解されたい。

【0044】上に図示かつ記載した例示的システムに鑑みて、本発明によって実施することができる方法は、図11および図12の流れ図を参照すると、よりよく理解されるであろう。説明を簡潔にするため、図11および図12の方法を一連のステップとして図示し、記載するが、本発明がこの順序のステップによって限定されず、いくつかのステップが、本発明によれば、異なる順序で、かつ／または、図示かつ本明細書に記載したもの以外のステップと同時に起こることができることを理解されたい。さらに、例示したすべてのステップが、本発明による方法を実施するために必要とされる可能性があるのではない。

【0045】ここで図11を参照して、参照カウントされたアイテムへの参照を獲得するための方法を流れ図にする。ステップ400で、全体的な初期化が実行される。たとえば、高速参照アイテムを作成することができ、参照カウントされたオブジェクトを作成することができ、参照カウントを、参照カウントされたオブジェクトにおいて確立することができ、参照カウント（たとえば、高速参照カウント（FRC））に関連付けられたローカル値を確立することができる。ステップ410で、ステップ400の参照カウントされたアイテムへの参照を獲得することを求めるアイテムが、FRCを検査して、ステップ400の高速参照アイテムが、獲得のために使用可能ないずれかの参照を有するかどうかを決定す

る。

【0046】ステップ420で、FRCが所定のしきい値より大きいかどうかに関する決定が行われる。ステップ420で決定がNOであった場合、ステップ430で、ロック、アクセスおよびアンロックを使用して参照を獲得するための従来の方法が実行される。ステップ440で、さらにいずれかの獲得者がいるかどうかに関する決定が行われる。ステップ440の決定がYESであった場合、制御がステップ410へ戻り、そうでない場合はこの方法が終了する。ステップ420の決定がYESであった場合、ステップ450で、獲得アイテムが高速参照アイテムであるかどうかに関する決定が行われる。ステップ450の決定がNOであった場合、ステップ460で、非高速参照アイテムが、ステップ400の参照カウントされたアイテムへの1つの参照を、アトミックオペレーションを使用してFRCを操作することにより獲得し、その後、制御がステップ440へ移る。ステップ450の決定がYESであった場合、ステップ470で、獲得高速参照アイテムが1個の参照を獲得し、1は整数である。ステップ470で、アトミックオペレーションが使用されて、参照の獲得元のアイテムのFRCが操作される。

【0047】ステップ480で、ステップ470の獲得により、参照の獲得元のアイテムにおけるFRCがゼロにされたかどうかに関する決定が行われる。ステップ480の決定がNOであった場合、制御がステップ440へ移る。しかし、ステップ480の決定がYESであった場合、ステップ490で、獲得高速参照アイテムが、参照の獲得元の高速参照アイテムのFRCを元通りにしようとする可以尝试することができる。たとえば、獲得高速参照アイテムが、ステップ400の参照カウントされたアイテムの参照カウントを検査し、追加の参照が使用可能かどうかを決定することができる。このような追加の参照が使用可能であった場合、獲得高速参照アイテムが、ロック、アクセスおよびアンロック方法を使用して、追加の参照を獲得することができ、追加の参照を、FRCがゼロにされた高速参照アイテムに割り当てることができる。別法として、獲得高速参照アイテムが追加の参照を保持することができ、これは続いて他の獲得アイテムによる獲得のために使用可能となる。

【0048】ここで、図12を参照して、参照カウントされたアイテムへの1つまたは複数の参照を解放するための方法を流れ図にする。ステップ500で、1つまたは複数の参照を解放するアイテムが高速参照アイテムかどうかに関する決定が行われる。ステップ500の決定がYESであった場合、ステップ510で、解放アイテムが1つまたは複数の参照を、それが参照を獲得した元の高速参照アイテムへ戻したいかどうかに関する決定が行われる。ステップ510の決定がNOであった場合、ステップ520で、解放アイテムが、参照カウントされ

たアイテムを直接デリファレンスし、その後、制御がステップ530へ移る。ステップ530で、さらにいずれかの解放アイテムがあるかどうかに関する決定が行われる。ステップ530の決定がYESであった場合、制御がステップ500へ移り、そうでない場合はこの方法が終了する。ステップ510で、決定がYESであった場合、ステップ540で、参照の獲得元の高速参照アイテムがなお存在するかどうかに関する決定が行われる。ステップ540の決定がNOであった場合、制御がステップ520へ移る。そうでない場合、ステップ540の決定がYESであった場合、ステップ550で、復帰アイテム (returning item) が1個の参照を戻し、1は1以上の整数である。復帰アイテムが、高速参照アイテムから獲得した参照のすべて、あるいはすべてより少ないものを戻すことができる。ステップ560で、解放アイテムが、参照カウントされたアイテムへのいずれかの参照を残しているかどうかに関する決定が行われる。決定がNOであった場合、解放アイテムがもはや、参照カウントされたアイテムを参照せず、ステップ570で、解放アイテムのFRCをNULL値に設定し、これがもはや、参照カウントされたアイテムを参照しないことを指示し、その後、制御がステップ530へ移る。ステップ560の決定がYESであった場合、解放アイテムがなお、参照カウントされたアイテムを参照し、したがって制御がステップ530へ移る。

【0049】ステップ500の決定がNOであった場合、解放アイテムが高速参照アイテムではなく、ステップ580で、解放アイテムがその参照を、それが参照を獲得した元の高速参照アイテムへ戻したいかどうかに関する決定が行われる。ステップ580の決定がNOであった場合、ステップ590で、解放アイテムが、参照カウントされたアイテムを直接デリファレンスし、その後、制御がステップ530へ移る。ステップ580で、決定がYESであった場合、ステップ600で、参照の獲得元の高速参照アイテムがなお存在するかどうかに関する決定が行われる。ステップ600の決定がNOであった場合、制御がステップ590へ移る。そうでない場合、ステップ600の決定がYESであった場合、ステップ610で、復帰アイテムがその参照を、それが参照を獲得した元の高速参照アイテムへ戻し、その後、制御がステップ530へ移る。

【0050】図13を参照すると、本発明の様々な態様を実施するための例示的環境710が、コンピュータ712を含み、これは、処理装置714、システムメモリ716、およびシステムバス716を含み、これが、システムメモリ716を含む様々なシステム構成要素を処理装置714へ結合する。処理装置714は、様々な市販のプロセッサのいずれにすることもできる。デュアルマイクロプロセッサおよび他のマルチプロセッサアーキテクチャを、処理装置714として使用することもでき

る。

【0051】システムバス716を、いくつかのタイプのバス構造のいずれにすることもでき、これには、様々な市販のバスアーキテクチャのいずれかを使用した、メモリバスまたはメモリコントローラ、周辺バス、ローカルバスが含まれる。システムメモリ716は、読み取り専用メモリ (ROM) 720およびランダムアクセスメモリ (RAM) 722を含む。基本入出力システム (BIOS) は、起動中など、情報をコンピュータ712内の要素の間で転送するための助けとなる基本ルーチンを含み、ROM 720に格納される。

【0052】コンピュータ712はさらに、ハードディスクドライブ724、たとえば、取外し可能ディスク728から読み取りあるいはこれに書き込むための磁気ディスクドライブ726、および、たとえば、CD-ROMディスク732を読み取るか、あるいは他の光媒体から読み取りあるいはこれに書き込むための光ディスクドライブ730を含む。ハードディスクドライブ724、磁気ディスクドライブ726、および光ディスクドライブ730が、システムバス716へ、それぞれハードディスクドライブインターフェイス734、磁気ディスクドライブインターフェイス736、および光ドライブインターフェイス738によって接続される。ドライブおよびそれらの関連付けられたコンピュータ可読媒体が、適切なデジタルフォーマットによるブロードキャストプログラミングの格納用を含む、コンピュータ712用の、データ、データ構造、コンピュータ実行可能命令などの不揮発性記憶装置を提供する。上のコンピュータ可読媒体の説明では、ハードディスク、取外し可能磁気ディスクおよびCDを参照するが、zipドライブ、磁気カセット、フラッシュメモ리카ード、デジタルビデオディスク、ペルヌーイカートリッジなど、コンピュータにより可読である他のタイプの媒体も例示的動作環境において使用できること、および、さらに、いかなるこのような媒体も、本発明の方法を実行するためのコンピュータ実行可能命令を含むことができることを、当業者には理解されたい。

【0053】いくつかのプログラムモジュールをドライブおよびRAM 722に格納することができ、これには、オペレーティングシステム740、一つまたは複数のアプリケーションプログラム742、他のプログラムモジュール744およびプログラムデータ746が含まれる。例示したコンピュータにおけるオペレーティングシステム740は、たとえば、「Microsoft (登録商標) Windows (登録商標) NT」オペレーティングシステムであるが、本発明を、様々な市販のオペレーティングシステム、またはオペレーティングシステムの組み合わせにより実施できることを理解されたい。

【0054】ユーザが、コマンドおよび情報をコンピュ

ータ712へ、キーボード748、および、マウス750などのポインティングデバイスを介して入力することができる。他の入力デバイス（図示せず）には、マイクロフォン、IRリモートコントロール、ジョイスティック、ゲームパッド、衛星放送受信アンテナ、スキャナなどが含まれる可能性がある。これらおよび他の入力デバイスが、しばしば処理装置714へ、システムバス716に結合されたシリアルポートインターフェイス752を介して接続されるが、パラレルポート、ゲームポート、ユニバーサルシリアルバス（「USB」）、IRインターフェイスなど、他のインターフェイスによって接続することもできる。モニタ754または他のタイプの表示デバイスも、システムバス716へ、ビデオアダプタ756などのインターフェイスを介して接続される。モニタ754に加えて、コンピュータ712は典型的には、スピーカ、プリンタなど、他の周辺出力デバイス（図示せず）を含む。

【0055】コンピュータ712は、ネットワーク環境において、リモートコンピュータ758など、1つまたは複数のリモートコンピュータへの論理接続を使用して動作することができる。リモートコンピュータ758は、ワークステーション、サーバコンピュータ、ルータ、パーソナルコンピュータ、マイクロプロセッサに基づいた娯楽電気器具、ピアデバイス、または他の共通ネットワークノードにすることができ、典型的には、コンピュータ712に関して記載した要素の多数またはすべてを含むが、簡潔にするため、記憶デバイス760のみを例示する。図示の論理接続は、ローカルエリアネットワーク（LAN）762および広域ネットワーク（WAN）764を含む。このようなネットワーク環境は、オフィス、企業全体のコンピュータネットワーク、イントラネットおよびインターネットにおいて一般的である。

【0056】LANネットワーク環境において使用するとき、コンピュータ712がローカルネットワーク762へ、ネットワークインターフェイスまたはアダプタ766を介して接続される。WANネットワーク環境において使用するとき、コンピュータ712が典型的にはモデム768を含み、あるいは、LAN上の通信サーバに接続され、あるいは、インターネットなどのWAN764を介して通信を確立するための他の手段を有する。モデム768を内部あるいは外部にすることができ、システムバス716へ、シリアルポートインターフェイス752を介して接続して、たとえば、POTSを介して通信を可能にすることができる。代替実施形態では、モデム768を内部あるいは外部にすることができ、システムバス716へ、ネットワークアダプタ766を介して接続して、たとえば、DSLおよび/またはモデムを介して通信を可能にすることができる。ネットワーク環境では、コンピュータ712に関して示したブ

ログラムモジュール、またはその一部を、リモート記憶デバイス760に格納することができる。図示のネットワーク接続は例示的であり、通信リンクをコンピュータの間で確立する他の手段を使用できることを理解されたい。

【0057】上に記載したものは、本発明の例示を含むものである。もちろん、考えられるあらゆる組み合わせの構成要素または方法を、本発明を説明するために記載することは可能ではないが、本発明の多数のさらなる組み合わせおよび交換が可能であることは、当業者には理解されよう。したがって、本発明は、添付の特許請求の範囲の技術思想および技術範囲の中に含まれるそのような変更、修正および変形形態をすべて包含することを意図するものである。さらに、「含む（include）」という用語が詳細な説明または特許請求の範囲において使用される範囲まで、このような用語が、「具備する（comprising）」という用語が請求項における従来の言葉として解釈されることに類似した方法において、包含的であることを意図するものである。

【図面の簡単な説明】

【図1】本発明の一実施形態による、アクセスされている、参照カウントされたアイテムを例示する概略的ブロック図である。

【図2】本発明の一実施形態による、高速参照アイテムに関連付けられた高速参照カウントを例示する概略的ブロック図である。

【図3】本発明の一実施形態による、参照カウントされたアイテムへのポインタに格納されている、参照カウントされたアイテムにおける参照カウントに関連付けられたローカル値を例示する図である。

【図4】本発明の一実施形態による、在来ポインタ、および、参照カウントに関連付けられたローカル値を格納するように適合されたポインタを例示する図である。

【図5】本発明の一実施形態による、参照カウントに関連付けられたローカル値を格納するように適合された、別のポインタを例示する図である。

【図6】本発明による、参照カウントされたアイテムにおける参照カウント、および、参照アイテムにおける参照カウントに関連付けられたローカル値を例示するタイミング図である。

【図7】本発明による、参照カウントされたアイテムにおける参照カウント、および、参照アイテムにおける参照カウントに関連付けられたローカル値を例示する、別のタイミング図である。

【図8】本発明による、参照カウントされたアイテムにおける参照カウント、および、参照アイテムにおける参照カウントに関連付けられたローカル値を例示する、別のタイミング図である。

【図9】本発明による、参照カウントされたアイテムにおける参照カウント、および、参照アイテムにおける参

照カウントに関連付けられたローカル値を例示する、別のタイミング図である。

【図 10】本発明の一実施形態による、サンプルデータフローを例示するデータフロー図である。

【図 11】本発明の一実施形態による、参照カウントされたアイテムへの参照を獲得するための方法を例示する流れ図である。

【図 12】本発明の一実施形態による、参照カウントされたアイテムへの参照を解放するための方法をさらに例示する流れ図である。

【図 13】本発明によって構成されたシステムにおける例示的動作環境の概略的ブロック図である。

【符号の説明】

10 システム

12、70、74、114、150、200、204、300、310 高速参照アイテム

14、116、154、320 非高速参照アイテム

15 新しいアイテム

16、72、112、152、202、302 参照カウントされたアイテム

30、90、130、170、220、304、308、312、322、324、330 参照カウント

31 X参照カウント

32、52、60、62、64、66、80、100、120、160、223、230、306、314 高速参照カウント (FRC)

33 F参照カウント

34、40、50、68 ポインタ

35 高速参照カウント作成器

42 変数

714 処理装置

716 システムメモリ

718 バス

720 ROM

722 RAM

724 ハードドライブ

726 磁気ディスクドライブ

728 ディスク

10 730 CD-ROMドライブ

732 ディスク

734 インターフェイス

736 インターフェイス

738 インターフェイス

740 オペレーティングシステム

742 アプリケーション

744 モジュール

746 データ

748 キーボード

20 750 マウス

752 シリアルポートインターフェイス

754 モニタ

756 ビデオアダプタ

758 リモートコンピュータ

760 記憶装置

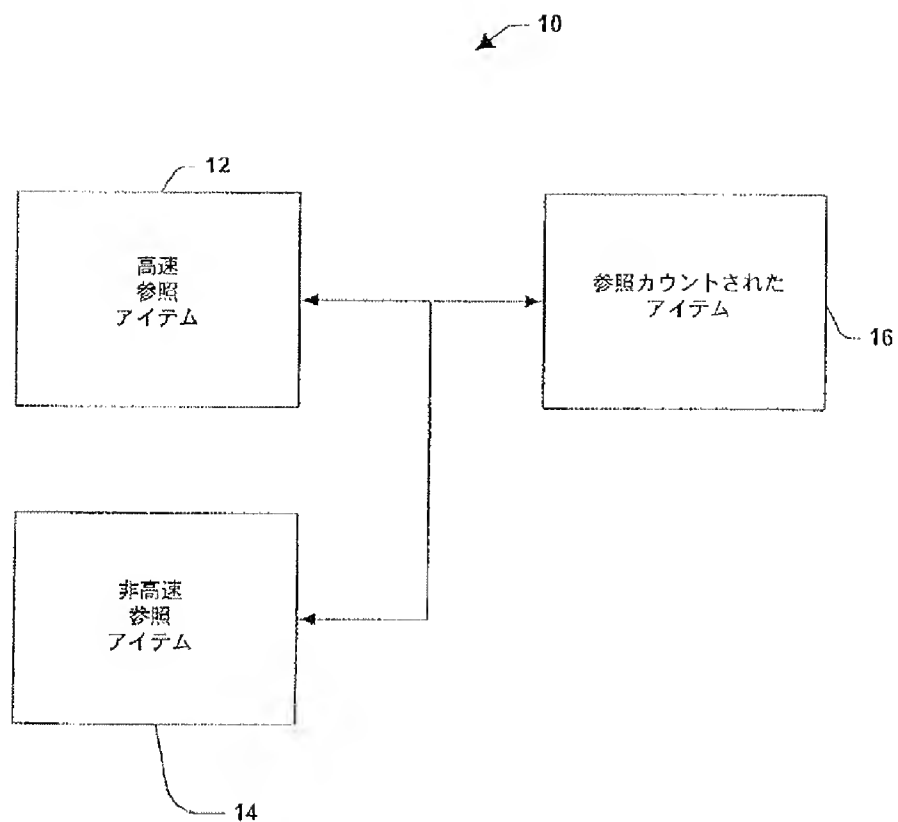
762 LAN

764 WAN

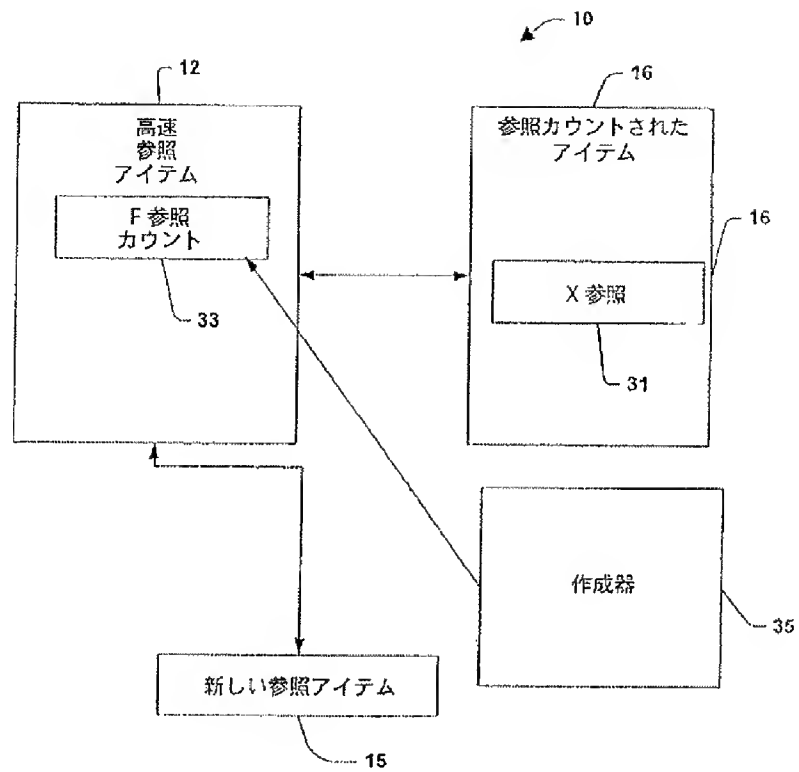
766 ネットワークアダプタ

768 モデム

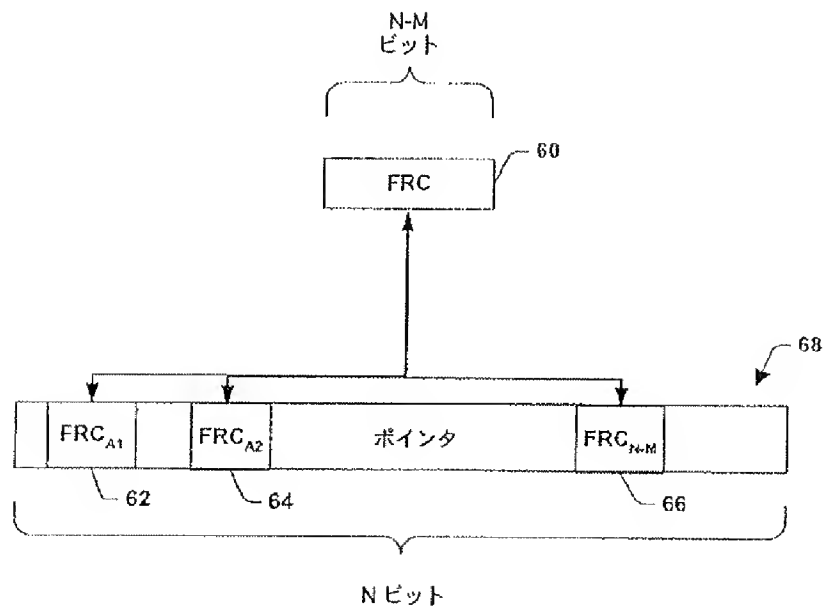
【図1】



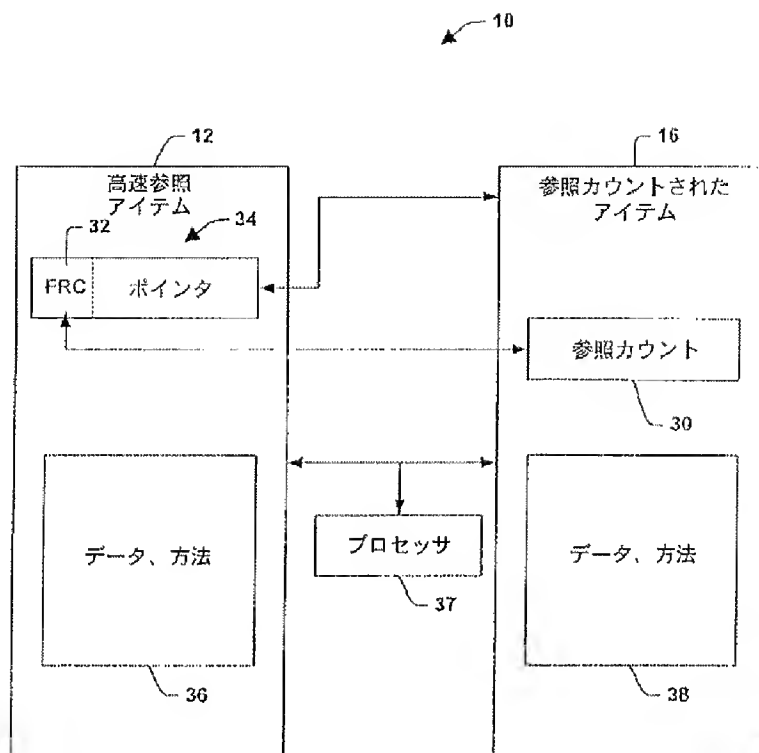
【図2】



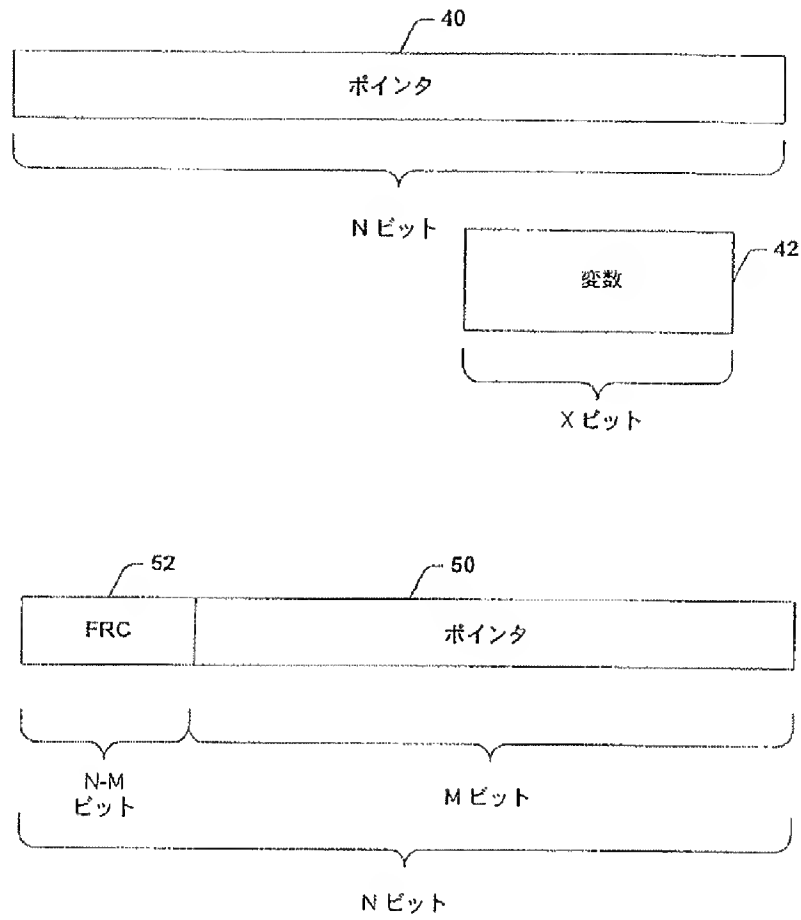
【図5】



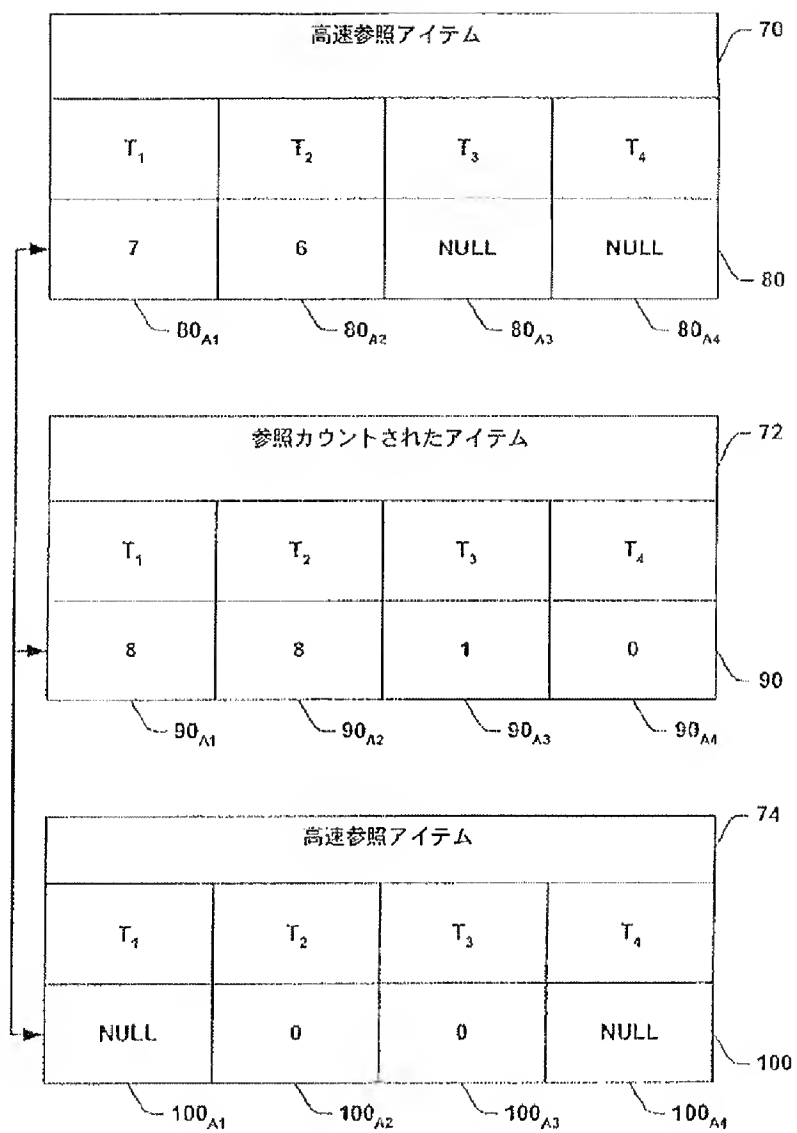
【図3】



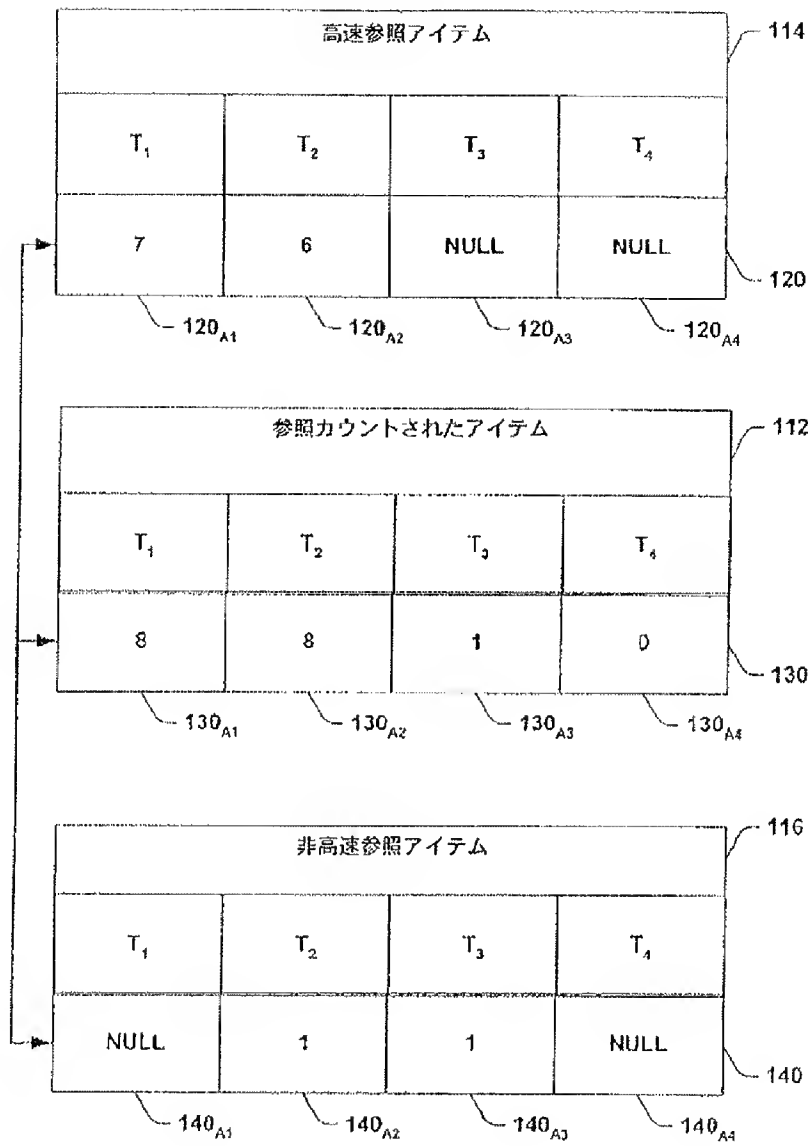
【図4】



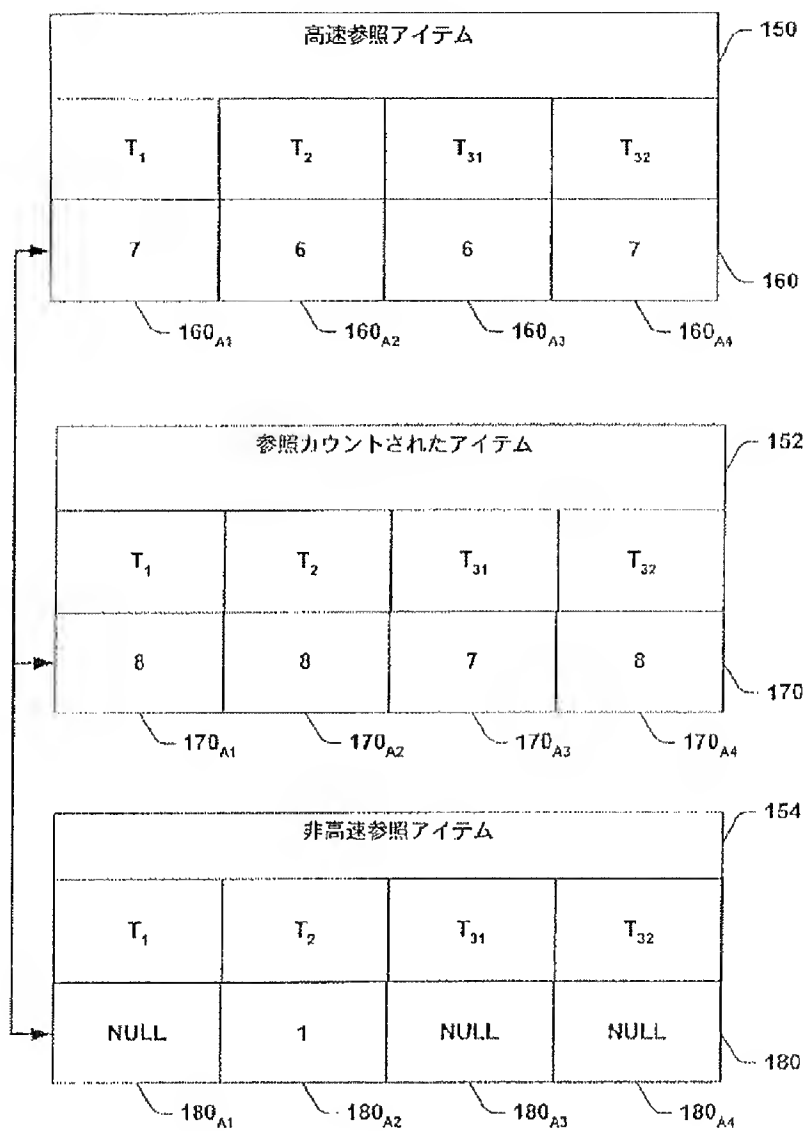
【図6】



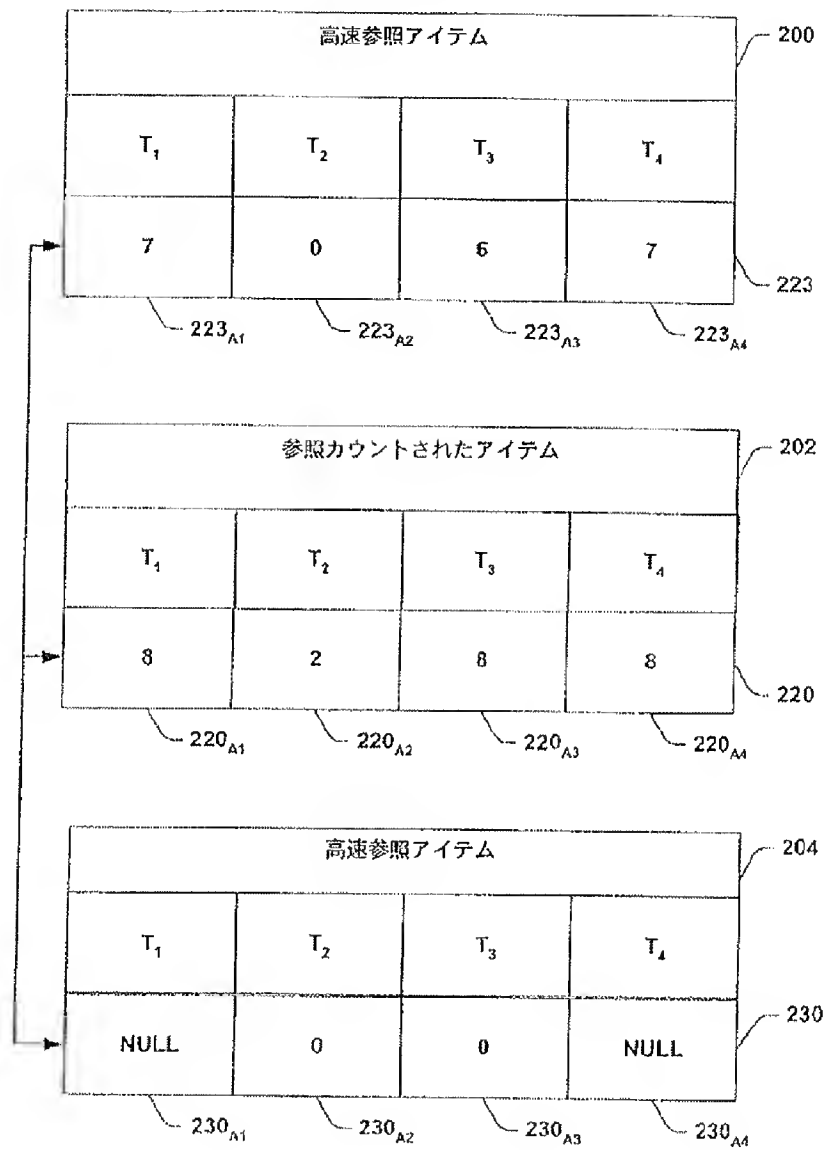
【図7】



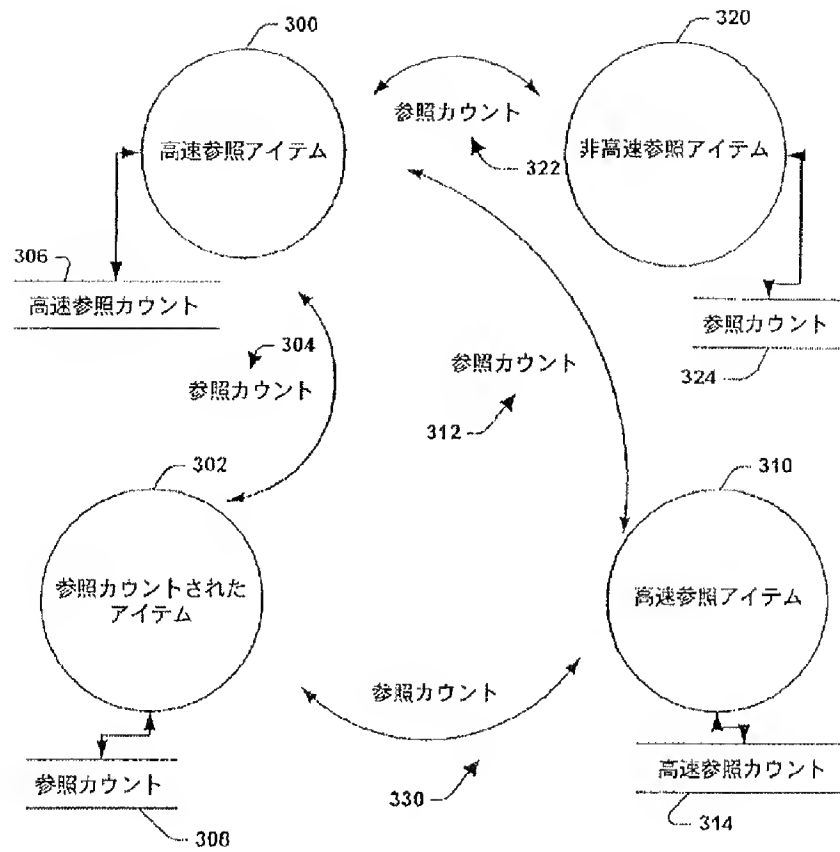
【図8】

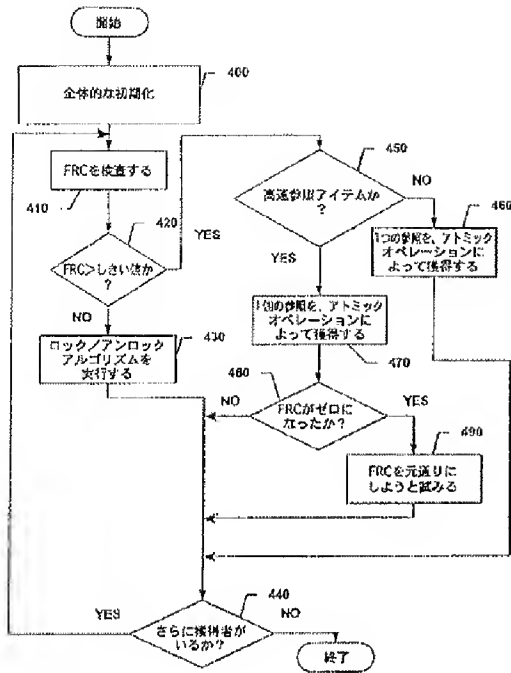


【図9】

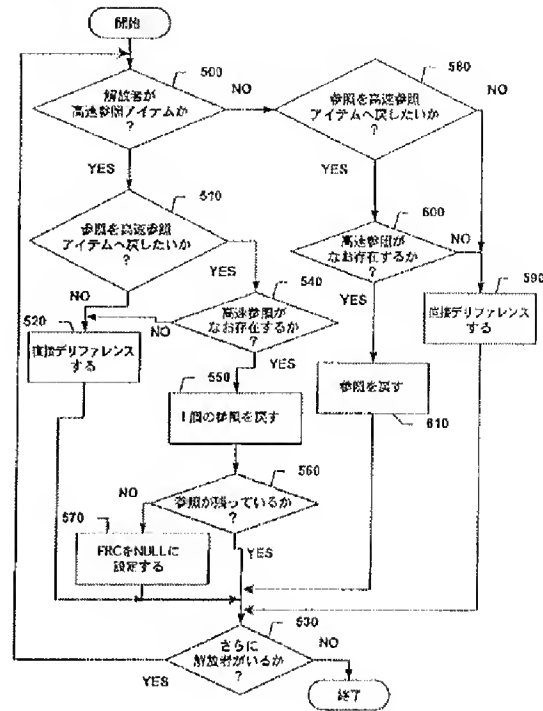


【図10】

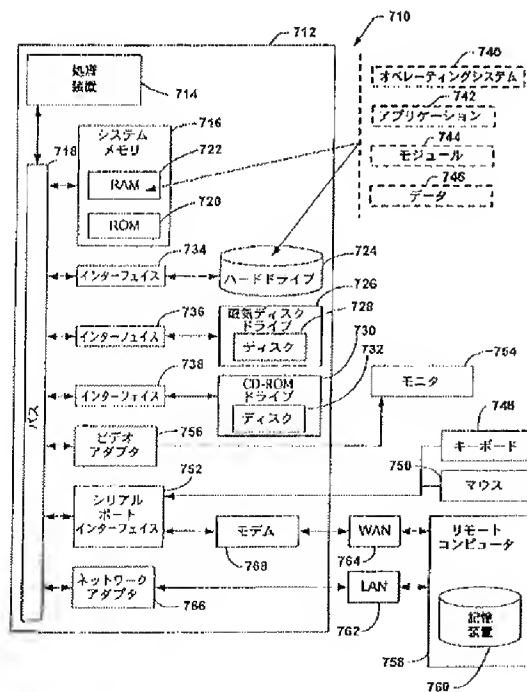


$$\begin{bmatrix} 1 & 1 \end{bmatrix}$$


【图 1-2】



【例】 3】



フロントページの続き

ドタム(参考) 5B060 AA07 CD17
5B098 AA03 AA10 GA05 GD03 GD15
GD22

